

WHAT IS CLAIMED IS:

- 1 1. A method comprising:
 - 2 calculating, in parallel, a first multiplication product of a first coefficient and a
 - 3 first sample, and a second multiplication product of the first coefficient
 - 4 and a second sample; and
 - 5 wherein the first sample and the second sample are from a plurality of sequential
 - 6 samples;
 - 7 wherein the first sample is an (n)th sample and the second sample is an (n+2)th
 - 8 sample in the plurality of sequential samples.
- 9 2. The method as recited in Claim 1, further comprising:
 - 10 full scale negative testing the first sample and the second sample.
- 11 3. The method as recited in Claim 1, further comprising:
 - 12 accumulating subsequent multiplication products with the first multiplication
 - 13 product into a final product.
- 14 4. The method as recited in Claim 3, further comprising:
 - 15 saturating the final product.
- 16 5. The method as recited in Claim 1, wherein the first sample and second sample
17 are odd samples in the plurality of sequential samples.
- 18 6. The method as recited in Claim 1, wherein the first sample and second sample
19 are even samples in the plurality of sequential samples.

20 7. The method as recited in Claim 1, further comprising:
21 calculating, in parallel, a third multiplication product of a second coefficient and a
22 third sample and a fourth multiplication product of the second coefficient
23 and a fourth sample;
24 wherein the third sample and the fourth sample are from the plurality of
25 sequential samples;
26 wherein the third sample is an (n+1)th sample and the second sample is an
27 (n+3)th sample in the plurality of sequential samples

28 8. The method as recited in Claim 1, further comprising:
29 calculating, in parallel a fifth multiplication product of a second coefficient and
30 the first sample, and a sixth multiplication product of the second
31 coefficient and the second sample generating a sixth product, and
32 accumulating in parallel, the fifth multiplication product with the first
33 multiplication product and the sixth multiplication product with the second
34 multiplication product.

35 9. The method as recited in Claim 5, wherein the first coefficient and second
36 coefficient are filter coefficients.

37 10. The method as recited in Claim 1, wherein the calculating in parallel
38 comprises executing a multiply accumulate single-instruction-multiple-data (SIMD)
39 instruction.

40 11. A method comprising:

41 calculating a finite impulse response (FIR), wherein the calculating includes
42 executing, in parallel, a first multiply accumulate operation of a first
43 sample with a first coefficient and a second multiply accumulate operation
44 of a second sample with the first coefficient;
45 wherein the first sample and the second sample are from a plurality of sequential
46 samples;
47 wherein the first sample is an (n)th sample and the second sample is an (n+2)th
48 sample in the plurality of sequential samples.

49 12. The method as recited in Claim 11, wherein the executing, in parallel,
50 comprises:
51 selecting the first sample in a first location of an upper half of a first register and
52 selecting the second sample in a corresponding location of a lower half of
53 the first register;
54 selecting a first operand in a first location of an upper half of a second register and
55 selecting a second operand in a corresponding location of a lower half of
56 the second register, wherein the first operand and the second operand are
57 each the first coefficient;
58 multiplying the first sample and the first operand and accumulating a first result in
59 a lower half of a third register; and
60 multiplying the second sample and the second operand and accumulating a second
61 result in an upper half of the third register.

62 13. The method as recited in Claim 12, further comprising saturating the first
63 result and the second result.

64 14. The method as recited in Claim 12, wherein the first location and the
65 corresponding location of the first register are one of the upper bits of each half of the
66 first register and the lower bits of each half of the first register.

67 15. The method as recited in Claim 11, wherein the first coefficient is a filter
68 coefficient.

69 16. The method as recited in Claim 11, wherein the calculating in parallel
70 comprises executing a multiply accumulate single-instruction/multiple-data (SIMD)
71 instruction.

72 17. An apparatus comprising:
73 a first plurality of multiplexers to select a first sample in a first location of an
74 upper half of a first register and a second sample in a corresponding
75 location of a lower half of the first register;
76 a second plurality of multiplexers to select a first operand in a first location of an
77 upper half of a second register and a second operand in a corresponding
78 location of a lower half of the second register, wherein the first operand
79 and the second operand are each a first coefficient;
80 a first multiplier to multiply the first sample with the first operand; and
81 a second multiplier to multiply the second sample with the second operand;

82 wherein the first sample and the second sample are from a plurality of sequential
83 samples;

84 wherein the first sample is an (n)th sample and the second sample is an (n+2)th
85 sample in the plurality of sequential samples.

86 18. The apparatus as recited in Claim 14, further comprising:
87 a first accumulator to accumulate a first product of the first multiplier; and
88 a second accumulator to accumulate a second product of the second multiplier.

89 19. The apparatus as recited in Claim 14, where a first selection control for the
90 first plurality of multiplexers and a second selection control for the second plurality of
91 multiplexers is according to a first qualifier and a second qualifier specified in a single-
92 instruction/multiple-data (SIMD) instruction.

93 20. The apparatus as recited in Claim 14, wherein the first coefficient is a filter
94 coefficient.

95 21. An apparatus comprising:
96 a first plurality of multiplexers to select a first sample in a first location of an
97 upper half of a first register and a second sample in a corresponding
98 location of a lower half of the first register;
99 a second plurality of multiplexers to select a first operand in a first location of an
100 upper half of a second register and a second operand in a corresponding
101 location of a lower half of the second register, wherein the first operand
102 and the second operand are each a first coefficient;

103 a first multiplier to multiply the first sample with the first operand; and
104 a second multiplier to multiply the second sample with the second operand;
105 wherein the first sample and the second sample are from a plurality of sequential
106 samples;
107 wherein the first sample is an (n)th sample and the second sample is an (n+2)th
108 sample in the plurality of sequential samples.

109 22. The apparatus as recited in Claim 21, the apparatus further comprising:
110 a first accumulator to accumulate a first product of the first multiplier; and
111 a second accumulator to accumulate a second product of the second multiplier.

112 23. The apparatus as recited in Claim 21, where a first selection control for the
113 first plurality of multiplexers and a second selection control for the second plurality of
114 multiplexers is according to a first qualifier and a second qualifier specified in a single-
115 instruction/multiple-data (SIMD) instruction.

116 24. The apparatus as recited in Claim 21, wherein the first coefficient is a filter
117 coefficient.

118 25. A data processing system comprising:
119 an addressable memory to store an instruction for a multiply-accumulate
120 operation;
121 a processing core coupled to the addressable memory, the processor core
122 comprising:
123 an execution core to access the instruction;

124 a first source register to store a plurality of sequential samples;

125 a second source register to store a plurality of coefficients; and

126 a destination register to store a plurality of results;

127 a wireless interface to receive data; and

128 an I/O system and decoder to provide the plurality of samples to the first source

129 register from the data;

130 wherein the execution core comprises:

131 a first plurality of multiplexers to select a first sample in a first

132 location of an upper half of a first register and a second sample in a

133 corresponding location of a lower half of the first register;

134 a second plurality of multiplexers to select a first operand in a first

135 location of an upper half of a second register and a second operand in a

136 corresponding location of a lower half of the second register, wherein the

137 first operand and the second operand are each a first coefficient;

138 a first multiplier to multiply the first sample with the first operand;

139 and

140 a second multiplier to multiply the second sample with the second

141 operand;

142 wherein the first sample is an (n)th sample and the second sample is an (n+2)th

143 sample in the plurality of sequential samples.

144 26. The data processing system as recited in Claim 25, the execution unit further

145 comprising:

146 a first accumulator to accumulate a first product of the first multiplier; and

147 a second accumulator to accumulate a second product of the second multiplier.

148 27. The data processing system as recited in Claim 25, where a first selection
149 control for the first plurality of multiplexers and a second selection control for the second
150 plurality of multiplexers is according to a first qualifier and a second qualifier specified in
151 a single-instruction/multiple-data (SIMD) instruction.

152 28. The data processing system as recited in Claim 25, wherein the first
153 coefficient is a filter coefficient.

154 29. An article comprising a storage medium having instructions stored thereon,
155 the instructions operable to:

156 calculate, in parallel, a first multiplication product of a first coefficient and a first
157 sample, and a second multiplication product of the first coefficient and a
158 second sample;

159 wherein the first sample and the second sample are from a plurality of sequential
160 samples;

161 wherein the first sample is an (n)th sample and the second sample is an (n+2)th
162 sample in the plurality of sequential samples.

163 30. The article as recited in Claim 29, wherein the first sample and second sample
164 are odd samples in the plurality of sequential samples.

165 31. The article as recited in Claim 29, wherein the first sample and second sample
166 are even samples in the plurality of sequential samples.

167 32. The article as recited in Claim 29, the instructions further operable to:

168 calculate, in parallel, a third multiplication product of a second coefficient and a
169 third sample and a fourth multiplication product of the second coefficient
170 and a fourth sample;

171 wherein the third sample and fourth sample are from the plurality of sequential
172 samples;

173 wherein the third sample is an (n+1)th sample and the second sample is an
174 (n+3)th sample in the plurality of sequential samples

175 33. The article as recited in Claim 29, the instructions further operable to:

176 calculate, in parallel a fifth multiplication product of a second coefficient and the
177 first sample, and a sixth multiplication product of the second coefficient
178 and the second sample generating a sixth product, and
179 accumulate in parallel, the fifth multiplication product with the first multiplication
180 product and the sixth multiplication product with the second multiplication
181 product.

182 34. The article as recited in Claim 33, wherein the first coefficient and second
183 coefficient are filter coefficients.

184 35. The article as recited in Claim 29, wherein to calculate in parallel comprises to
185 execute a multiply accumulate single-instruction-multiple-data (SIMD) instruction.